

## Homework 2

### Problem 2: Linear separability of 4-dimensional Boolean functions

For the first problem I wrote the following program in MATLAB:

```
inputData = csvread('input_data_numeric.csv');
inputData(:,1)=[];
inputDataSize = size(inputData,1);
updates = 10^5;
targets = [[1, 1, 1, -1, 1, 1, 1, 1, -1, -1, -1, -1, 1, 1, -1, -1]
           [1, 1, -1, -1, -1, -1, -1, 1, -1, -1, 1, 1, 1, -1, -1, 1]
           [-1, -1, -1, 1, -1, 1, 1, 1, 1, 1, 1, -1, 1, -1, -1, -1]
           [1, 1, -1, 1, -1, -1, 1, -1, 1, -1, 1, 1, -1, 1, -1, 1]
           [1, -1, 1, 1, 1, 1, -1, -1, 1, -1, 1, 1, 1, 1, -1, 1]
           [-1, 1, -1, -1, -1, -1, -1, 1, -1, -1, -1, -1, -1, 1, -1, -1]
          ];
learningRate = 0.02;
numOfWeights = size(inputData,2);
O = zeros(inputDataSize,1);

for numOfFunc = 1:size(targets,1)

    threshold = 2*rand-1; % [-1,1]
    w = 0.4*rand(numOfWeights, 1)-0.2; % [-0.2, 0.2]

    for j=1:updates
        mu = randi(inputDataSize);
        wSum = 0;
        for k = 1:size(inputData,2)
            wSum = wSum + w(k)*inputData(mu,k);
        end

        b = 1/2*(-threshold+wSum);
        O(mu) = tanh(b);
        if isequal(transpose(sign(O)), targets(numOfFunc,:))
            fprintf('Function %d is linear.\n', numOfFunc);
            break
        end

        wUpdate = learningRate*((targets(numOfFunc,mu)-O(mu))*(1-(tanh(b))^2)*...
            transpose(inputData(mu,:)));
        thresholdUpdate = -learningRate*((targets(numOfFunc, mu)-O(mu))*...
            ((1-tanh(b))^2));
        w = w + wUpdate;
        threshold = threshold + thresholdUpdate;

    end % num of updates

end % num of functions
```